Patent Office
Canberra

I, JULIE BILLINGSLEY, TEAM LEADER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. 2002951013 for a patent by [illegible]PPO KERONEN as filed on 27 August 2002.

WITNESS my hand this
Twenty-third day of July 2003

J. Billingsley

JULIE BILLINGSLEY
TEAM LEADER EXAMINATION
SUPPORT AND SALES

ABSTRACT OF THE DISCLOSURE

A method for delivering data services to users of portable terminals is disclosed. The new method offers lower cost, easier management, better security and better usability of data communications services than existing methods. The method employs a cached, compressed, monitored, persistent, secure and switched (CCMPSS) data tunnel between a portable terminal and a data communications gateway. The relevant aspects of the terminal and gateway apparatus that support the method are disclosed. Embodiments for a selection of products that employ the disclosed method and apparatus are described.

BACKGROUND OF THE INVENTION

Field of the Invention

Many data communications access networks are available now, many are under construction and many are planned for the future. These access networks provide the means for terminal devices to access data services hosted on the public internet and private intranet networks. Examples of terminal devices are notebook computers, tablet or notepad computers, personal digital assistant (PDA) devices and smart cellular phones. Examples of data services access methods and apparatus using access networks are telephone modem and DSL modem access via the public switched telephone network, cable modem access via coaxial and fiber cable networks, GSM/GPRS access via a cellular mobile telephone network and wireless modem access via an IEEE 802.11 wireless LAN access point.

Many access networks allow internet protocol (IP) data packets to be routed to the global internet infrastructure, which in turn is designed to route the packets to any desired internet host address.

Many private networks or intranets are connected to the internet by means of a firewall host computer. The firewall is designed to protect the privacy and functionality of the private intranet.

The infrastructure, described above, consisting of access networks, the public internet and private intranets, provide the basic means for terminal devices connected to the infrastructure to access data services hosted on server computers or peer terminal devices connected to the infrastructure. The new method and apparatus described in the present disclosure offers lower cost, easier management, better security, and better usability of data service access via the infrastructure than existing methods.

Description of the Related Art

Figure 1 illustrates the way that data communications infrastructure is currently used to access data services. Three separate data service access pathways are illustrated:

Host A: A terminal communicates with host computer A by means of an ordinary TCP/IP protocol connection or connectionless UDP/IP. The data communication packets are routed via an access network and intranet. Typically the computer hosting the service being accessed is connected to the internet via a high bandwidth access network. The illustrations in this document omit such details for clarity of explanation.

Host B: A terminal communicates with host computer B by means of a secure data communications protocol, such as IPSec or one of the many proprietary virtual private network (VPN) protocols available. Typically these protocols encapsulate IP protocol packets, hence the term *tunnel* is often used to refer to such secure connections.

Host C: A terminal communicates with host computer C located on a private intranet. The communications pathway is divided into a secured segment across the public networks and an unsecured segment across the intranet. This method enables the private data services hosted on an intranet to be accesses via insecure public networks.

There are several problems with the architecture illustrated in Figure 1. In order to solve these problems, we have invented a new method and associated apparatus for access to data services. We refer to this new system as the mobile Business Accelerator (mBA).

We now describe the main problems that the mBA system is designed to solve.

Security Related Problems

The security requirements for access to private services typically include the need to authenticate both the service client and the service provider in order to prevent access by unauthorized parties. Confidential information should not be available to unintended third parties. It is also typically required to keep verifiable account information of service access. Service availability should be ensured as far as possible, even under malicious service denial attacks.

The following specific security sub-problems are solved by the mBA system:

Service Context Authentication Problem: It may be required that access is only to be granted in specific circumstances, such as:

User Authentication: Access is only granted to a specific user, or in the presence of user or group pf users.

Device Authentication: It may be required that the terminal device and the service gateway are mutually authenticated.

Location Authentication: Specific services may only be available to terminals located at specific geographic locations.

Application Authentication: Malicious, faulty or incompatible software installed on an otherwise authorized terminal or gateway can present a security and service quality threat. Hence the presence of authorized software and absence of unauthorized software is often required.

Transport Security Problem: Data packets traversing a public access network and the internet are subject to interception and falsification. Interception by unauthorized parties presents an information privacy threat. Falsification of data packets presents privacy, service theft and service denial threats. For these reasons it is necessary to ensure that

intercepted data packets do not reveal private information to unauthorized parties, and to ensure that data packets that are injected into the network by unauthorized parties are detected and rejected.

File System Privacy Problem: In the case that either the terminal device or gateway device falls into the hands of unauthorized parties, any sensitive information stored on storage devices, such as disk drives, should be protected.

Non-repudiation problem: It may be required to keep a verifiable record of communications between terminals and a gateway. Such a record may be used to resolve any disputes arising between the service provider and service consumer.

Operating Cost Related Problems

The mBA system is designed to solve the following cost related problems:

High Cost Problem: Many access networks charge for network traffic based on traffic volume. Such charges are particularly high for wireless networks that employ licensed radio spectrum. These costs need to be minimized.

Cost Management Problem: It is often necessary to attribute costs to specific users or applications. Such information is typically used to manage costs and minimize future costs of operation.

Cost Auditing Problem: In order to eliminate disputes, verifiable records of network traffic may need to be available.

System Capacity Planning Problem: As more users drive more treaffic, connection bandwidth and data processing capacity need to be increased to meet the increased demand. Also if demand decreases, cost savings

may be possible by downsizing bandwidth and processing capacity. It is necessary to keep track of trends that indicate changed demand.

Connectivity Related Problems

The mBA system is designed to solve the following data communications connectivity related problems:

Long Latency Problem: Networks often employ a limited region of radio spectrum to offer a shared communications service to many clients. Contention for access to the shared communications medium means that long delays may occur between the time that the transfer of data is requested and the time when that transfer can actually take place. Applications that operate via such networks need to provide a responsive user experience, despite such adverse circumstances.

Network Coverage Problem: Wireless networks typically do not provide a uniform quality of coverage for an entire geographic region where access to the network is needed. There are often low signal strength and low signal quality areas, where the service bandwidth or reliability is reduced or where service is not available at all. Applications that are accessed via such networks need to operate as reliably as possible, despite such adverse circumstances.

Terminal Visibility Problem: Wireless and wired access networks often provide private internet protocol (IP) addresses, which are not visible to hosts on the internet. In this case an internet host, is not able to establish a connection to the terminal. This functional deficiency means that applications that require the connection to be established by a server or peer to the terminal cannot be used.

Multiple Networks Problem: More than one access network may be available at any one time or over a period of time. For example, the terminal may be able to communicate via a GPRS cellular network, a wireless LAN network and a fixed line Ethernet LAN. Where multiple access networks are available at the one time or as availability of access networks changes over time, the terminal user is currently called on to manually select which network is actually used.

Product Distribution Related Problems

The mBA system is designed to solve the following problems related to achieving wide, low cost distribution and customer acceptance of the product:

Packaging Problem: Data communications systems are complex and solutions often consist of many products each with many sub-components. A simple to deploy product that offers the whole solution to a data communication need is difficult to achieve.

Communications Gateway Deployment Problem: Data communications products that consist of two apparatus parts, a terminal and a gateway, typically require a wide deployment of the gateway component before the terminal component is sufficiently useful. This requirement makes the economic deployment of such products difficult.

SUMMARY OF THE INVENTION

mBA, the invention described in this disclosure is a system, associated methods and apparatus that solves the above problems in an efficient and novel way. The central new technology employed to achieve this result is a cached, compressed, monitored, persistent, secure, switched (CCMPSS) tunnel.

A network architecture view of the mBA system is presented in Figure 2. The system consists of two top-level pieces of apparatus, namely:

> **mBA Terminal**: This is any access terminal device provided with at least one network interface, and fitted with the mBA client method, embodied as a software, firmware or hardware implementation.
>
> **mBA Gateway**: This is a computer provided with at least one network interface, and fitted with the mBA gateway method, embodied as a software, firmware or hardware implementation.

The CMPSS tunnel is the logical connection and protocol used to transfer data between the terminal and gateway. As illustrated in Figure 2, the tunnel is generated and maintained by the mBA client method at the terminal end, and by the mBA gateway method at the gateway end.

Figure 2 illustrates the way that data communications infrastructure is employed by the mBA system to provide access to data services. For comparison with the existing art, the three data service access scenarios used in Figure 1, are illustrated in Figure 2. In this case using the mBA system the three data access paths are:

> **Host A**: A terminal communicates with host computer A by means of an ordinary TCP/IP protocol connection or connectionless UDP/IP. The data communication packets are encapsulated and transported to the mBA

gateway through the CMPSS tunnel. The mBA gateway method recovers the packets and routes them to host A.

Host B: A terminal communicates with host computer B by means of a secure data communications protocol, such as IPSec or one of the many proprietary virtual private network (VPN) protocols available. Again the data communication packets are encapsulated and transported to the mBA gateway through the CMPSS tunnel. The mBA gateway method recovers the packets and routes them to host B.

Host C: A terminal communicates with host computer C located on a private intranet. The communications pathway is divided into a secured segment across the public networks and an unsecured segment across the intranet. The secured segment consists of a CMPSS tunnel. The mBA gateway method recovers the packets and routes them to host C.

The mBA Client Method

Figure 3 illustrates the structure of the environment for software applications, running on a terminal device, as realized in the existing art. The components of this structure are:

User Interfaces: User applications typically present a visual or audio interface that enable users to interact with the application. This component may be absent in some basic service applications.

User Applications: These are the executing programs that provide the instructions that determine the behavior of the application.

Network Protocol Stacks: This component implements the network addressing, packet formatting, security, and other protocol logic required to communicate successfully over a network. Examples of network stacks are the TCP/IP and UDP/IP stacks in common use today.

Network Interfaces: The terminal is connected to one or more networks by means of the network interfaces.

When the mBA client method is employed the terminal architecture is different from that shown in Figure 3. The new top-level structure is shown in Figure 4. This top-level architecture is refined in Figure 5 to reveal the functional components that are preferred to implement the client method.

The following sections describe the new top-level and functional components:

mBA Terminal User Interface (UI): The user can view and control the functions of the mBA system using this interface. This UI displays the status of network interfaces, as well as statistics of traffic per interface, per unit time and per application or protocol port number. When positive user authentication is required, this UI is used to recognize the user credentials.

mBA Terminal Applications: These are the main applications that are specific to the mBA terminal:

Control Application: The main functions of this application are user authentication, enabling the user to manually switch network interfaces and restrict access to nominated expensive transports by applications that are not considered high value enough to use such expensive bandwidth.

Logging Application: This application is able to display network statistics via the UI and communicate any logging information that is not available otherwise to the mBA gateway.

Log Collection: This is a service/daemon component thar collects raw statistics for use by the logging application.

Secure File System: In order to protect the privacy of information stored on the terminal. It is preferred that the terminal be equipped with an

encrypted file system. In this case, persistent user authorization data and compression/decompression caches may be stored securely.

Caches: The efficient compression of data transmitted between the terminal and gateway relies on the presence of synchronized caches at both ends of the connection. It is useful to store backup copies of these caches to enable efficient compression at terminal startup time.

For efficient use of the caches for compression, duplicate sequences should not be stored, and sequences that occur frequently should be stored so that they can be referenced using short cache addresses. A person skilled in the art can readily implement such caches.

> **Compression Cache**: This cache contains sequences of bytes that have been transmitted to the gateway. More than one compression cache may be used. As examples, caches may be allocated per protocol port/application, per mime-type or per connection. In any case, the gateway employs a corresponding set of decompression caches that have identical content to the corresponding terminal compression cache at identical positions in the transmitted stream of bytes.
>
> **Decompression Cache**: This cache contains sequences of bytes that have been received from the gateway. More than one decompression cache may be used. As examples, caches may be allocated per protocol port/application, per mime-type or per connection. In any case, the gateway employs a corresponding set of compression caches that have identical content to the corresponding terminal decompression cache at identical positions in the transmitted stream of bytes.

mBA Protocol Stack: This is the central component that generates the CCMPSS tunnel at the terminal end. The functional components of the stack are:

Compression/Decompression: The most important compression method is to use the synchronized caches provided at the terminal and gateway to replace repeated sequences of bytes by cache references. The compression converts a byte sequence to be transmitted into a sequence of literal and cache reference tokens. Decompression recovers the original sequence by looking up cache references from the decompression cache.

User Level Security: This component optionally encrypts/decrypts and digitally signs/verifies signature for transmitted data. Many algorithms are known in the art for these operations. The novel step that is employed by this component is to hash a checksum of the synchronized cache as part of a shared secret between the terminal and gateway.

Network Interface Switching: This component monitors the status of available network interfaces to determine which interfaces are able to provide a communications path to the gateway. In the case that a less expensive or higher bandwidth connection is available, the newly available connection is used. Note that the most expensive connections should not be used to send probe packets to the gateway, in order to avoid unnecessary cost.

The mBA Gateway Method

Figure 7 illustrates the logical structure of a firewall/gateway as realized in current practice. There is just one component that did not already appear in relation to Figure 3:

Packet Router: This component receives data from network interfaces, typically after some processing by network protocol stacks. In the case of a tunneling protocol, the data is typically unencapsulated and forwarded

on without address translation. The outgoing data is written to a network interface, typically after some processing by network protocol stacks.

When the mBA gateway method is employed the gateway architecture is different from that shown in Figure 7. The new top-level structure is shown in Figure 8. This top-level architecture is refined in Figure 9 to reveal the functional components that are preferred to implement the gateway method.

The following sections describe the new top-level and functional components:

mBA Gateway User Interface (UI): The user can view and control the functions of the mBA system using this interface. This UI displays the status of network interfaces, as well as statistics of traffic per interface, per unit time and per application or protocol port number. This UI also allows the authorization/de-authorization of terminals and terminal users. In order to provide for remote management of the mBA gateway, this UI is preferably implemented as a remotely accessible interface, such as a secure HTML web interface for example.

mBA Gateway Applications: These are tke main applications that are specific to the mBA gateway:

Control Application: The main functions of this application are user authentication, and access control.

Authorization Database: The information required to authenticate and authorize terminals and users is maintained in this database.

Logging Application: This application is able to display network statistics via the UI and receive any statistical information that terminals send to the gateway.

Log Collection: This is a service/daemon component thar collects raw statistics for use by the logging application.

Report Generator: This component generates on-demand or periodic reports of user activity and communications volumes, estimated costs and any other information that may be useful for auditing, non-repudiation, capacity planning, applications re-engineering and other purposes.

mBA Packet Router: This component employs existing practices for packet routing, except that connections to target hosts are maintained active while a terminal is unreachable due to lack of network connectivity.

Caches: The gateway maintains separate complementary compression and decompression caches for each authorized terminal. The other features of these caches are as in the case of the mBA terminal.

mBA Protocol Stack: This is the central component that generates the CCMPSS tunnel at the gateway end. The functional components of the stack are:

Compression/Decompression: See explanation for mBA terminal above.

User Level Security: See explanation for mBA terminal above.

Network Interface Switching: This component listens for switched traffic from terminals. Following the recognition and binding of a terminal to a new IP address, the gateway interface switch simply passes a persistent terminal identifier to the higher levels of the stack.

The CCMPSS Tunnel

As illustrated in Figure 2, the mBA system transports data across the access network used by the terminal by means of a CCMPSS tunnel. The CCMPSS

tunnel is designed to overcome problems in access networks and in roaming within and across access networks.

The CCMPSS tunnel consists of the following

Caching: Data transferred between the terminal and gateway is cached at both ends. This shared information about transferred data is useful for compression and security purposes. One or more transmit caches and one or more receive caches are maintained on both the terminal and the gateway. The terminal transmit cache content is the same as the corresponding gateway receive cache, when the identical position in the data communication stream is processed. The gateway transmit cache is similarly synchronized with the corresponding terminal receive cache.

Compression: Given the existence of synchronized caches, it is possible to reduce the amount of data traffic by replacing any segment of data that occurs in the transmit cache by a reference or pointer to that cache entry. The larger the cache, the more compression can typically be achieved. A good caching algorithm keeps often referenced data without duplication.

Monitoring: The terminal and gateway keep statistical information about the amount of traffic transmitted and compressed. This information is useful to collect so that it can be presented indexed by time, user, and port or application.

Persistence: In the case that the data communication connection between the terminal and gateway is temporarily lost, the terminal maintains the appearance of a logical connection for the terminal application software. Similarly the gateway maintains the appearance of an intact logical connection towards the target host software.

Security: Data transferred between the terminal and gateway is encrypted and digitally signed to provide security. The synchronized caches can be

used to increase security by including a checksum computed over the cache as part of a shared secret between the terminal and gateway.

Switching: In the case that the terminal switching component detects that multiple access networks are available for the transfer of data between the terminal and gateway, the switching component transfers data using the most economical and/or highest available bandwidth means.

The design of an efficient CCMPSS protocol can be readily carried out by a person skilled in the art. Compressed data can be represented as a sequence of literal and cache reference tokens. Any monitoring data that needs to be exchanged between the terminal and gateway can be transported normally through the CCMPSS tunnel as application traffic. Security may be implemented with minimal protocol overhead using secret key technologies; Alternatively public key mechanisms may be used. Switching simply requires a control packet that identifies a new terminal to IP address binding to enable the gateway to recognize the new terminal IP address.

DETAILED DESCRIPTION OF EMBODIMENTS

Embodiment for the Corporate Market

An embodiment of the mBA system, aimed as a product for small, medium and large businesses, has been constructed. The embodiment consists of the mBA terminal method implemented for notebook and tablet computers running the Windows XP and Windows 2000 operating systems, and the mBA gateway method implemented for computers running the FreeBSD and Linux operating systems.

The initial terminal implementation design is shown in Figure 6. The implementation follows the more general pattern shown in Figure 5. However, the following implementation details are significant:

Virtual Network Interface: The mBA terminal protocol stack is implemented as a separate process. Applications access the stack through a Windows virtual network interface. The stack uses the normal Win32 network API to perform network transport operations. Note that this approach is not optimized for performance. Subsequent implementations will implement the mBA terminal protocol stack as a lower level Windows NDIS driver.

Connection Multiplexing/De-multiplexing: This embodiment employs a single compression cache and a single decompression cache at the terminal, and the corresponding pair of compression and decompression caches at the gateway. In order to ensure synchronization of cache access, multiple network connections are multiplexed into a single TCP/IP connection between the terminal and gateway. Subsequent implementations may employ multiple connections between the terminal and gateway, with more sophisticated synchronization.

Interface Switching: This embodiment is specifically aimed at reducing the cost of 2.5G wireless network access, such as provided by GPRS cellular networks. The interface switching algorithm uses the relatively expensive cellular wireless transport as the lowest priority default transport. It is assumed that the other interfaces provide less expensive, higher bandwidth service. Subsequent implementations may employ an interface policy file to direct switching.

The initial gateway implementation design is shown in Figure 10. The implementation follows the more general pattern shown in Figure 9. However, the following implementation details are significant:

Connection Multiplexing/De-multiplexing: This is the gateway complement of the terminal multiplexing/de-multiplexing component.

Connection Table: A table of active TCP connections is used to maintain live connections to target hosts.

Embodiment for the Consumer Market

An embodiment of the mBA system, aimed as a product for the consumer market, has been constructed. This embodiment consists of the same terminal implementations as the preceding embodiment for the corporate market. However, the packaging of the product as an off-the-shelf solution with incremental functionality and the use of a public mBA gateway are novel aspects of this embodiment.

The "solution in a box" packaging of the embodiment is depicted in Figure 11. Figure 11 illustrates (a) the product package, containing (b) a GPRS wireless PC card, (c) a GPRS over GSM cellular network SIM card, (d) a CD-ROM containing installation images of the mBA software, and (e) a user guide containing installation instructions.

Figure 12 illustrates the incremental functionality product configuration process. The process consists of the following 4 steps. Step (a) is required for initial product functionality. The remaining steps are optional and may be performed in any order. The installation steps are:

Step (a): Installation of the mBA terminal software on a notebook computer enables the notebook to connect to a public mBA gateway for access to internet services. This access path is illustrated in Figure 2 for host A. For this type of access the security layer of the mBA protocol stack may be bypassed.

Step (b): Installation of a connection wizard on a personal computer (PC) enables the public gateway to set up a virtual private network connection to the consumer's PC. This access path is illustrated in Figure 2 for host B. Given this setup, the consumer is able to securely access private information stored on the personal computer.

Step (c): Installation of a connection wizard on a PC connected to an intranet enables the intranet PC to set up a virtual private network connection to the public mBA gateway. In this case the HTTP protocol that can bypass the corporate firewall is used to carry the encrypted data. This enables the consumer to securely access private information on the intranet PC.

Step (d): Installation of the mBA gateway software on a corporate intranet gateway host computer enables the normal intranet access functionality of the mBA system. This access path is illustrated in Figure 2 for host C.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 illustrates the way that data communications infrastructure is currently used to access data services.

Figure 2 depicts the same data service access scenarios as Figure 1, but this tine employing the new mBA system.

Figure 3 illustrates the logical structure of a terminal application, as realized in current practice.

Figure 4 shows how the logical structure of a terminal application is changed, compared to Figure 3, when the mBA terminal method is employed.

Figure 5 displays the main functional components of an mBA terminal embodiment.

Figure 6 specifies the structure and functional components of an mBA terminal embodiment implemented for Windows XP notebook computers as the terminal.

Figure 7 illustrates the logical structure of a firewall/gateway as realized in current practice.

Figure 8 shows how the logical structure of a gateway is changed, compared to Figure 7, when the mBA gateway method is employed.

Figure 9 displays the main functional components of an mBA gateway embodiment.

Figure 10 specifies the structure and functional components of an mBA gateway embodiment implemented for FreeBSD Unix computers as the gateway.

Figure 11 illustrates (a) an mBA product package, containing (b) a GPRS wireless PC card, (c) a GPRS over GSM cellular network SIM card, (d) a CD-

ROM containing installation images of the mBA software, and (e) a user guide containing installation instructions.

Figure 12 illustrates the incremental functional product configuration process, where step (a) installs the mBA terminal software on a notebook computer, (b) installs a connection wizard on a personal computer, (c) installs a connection wizard on a PC connected to an intranet, and (d) installs the mBA gateway software on a corporate intranet gateway host computer.

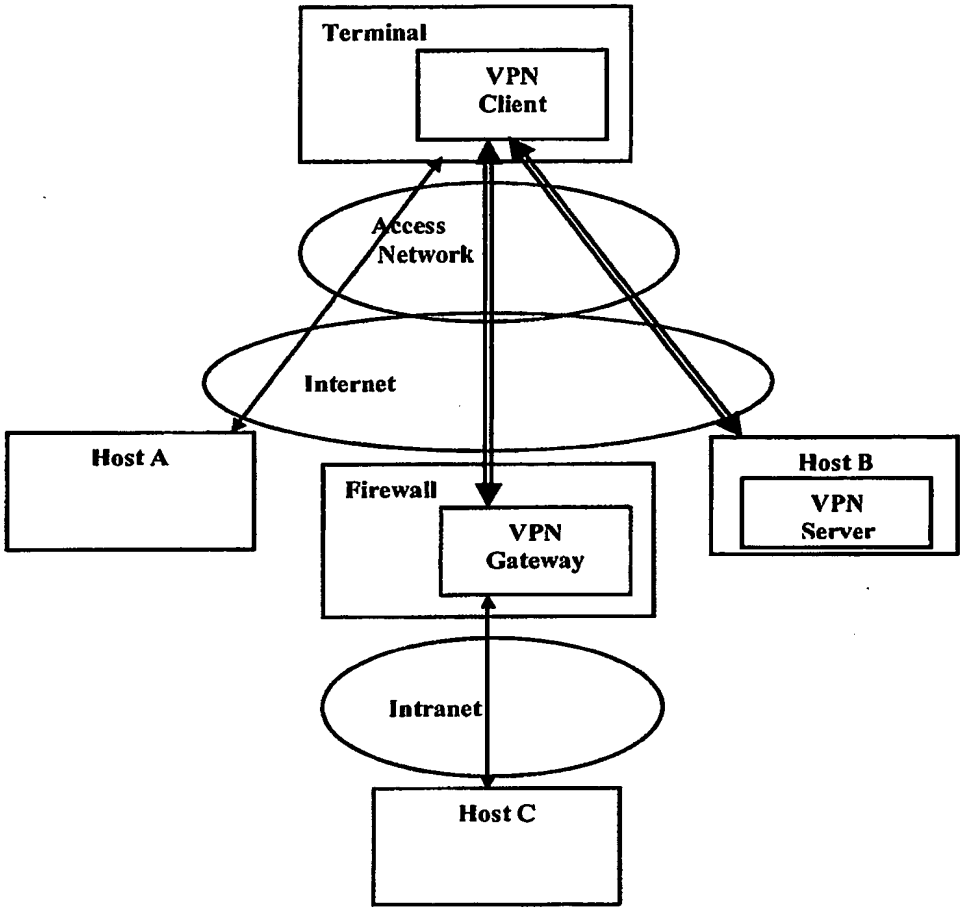
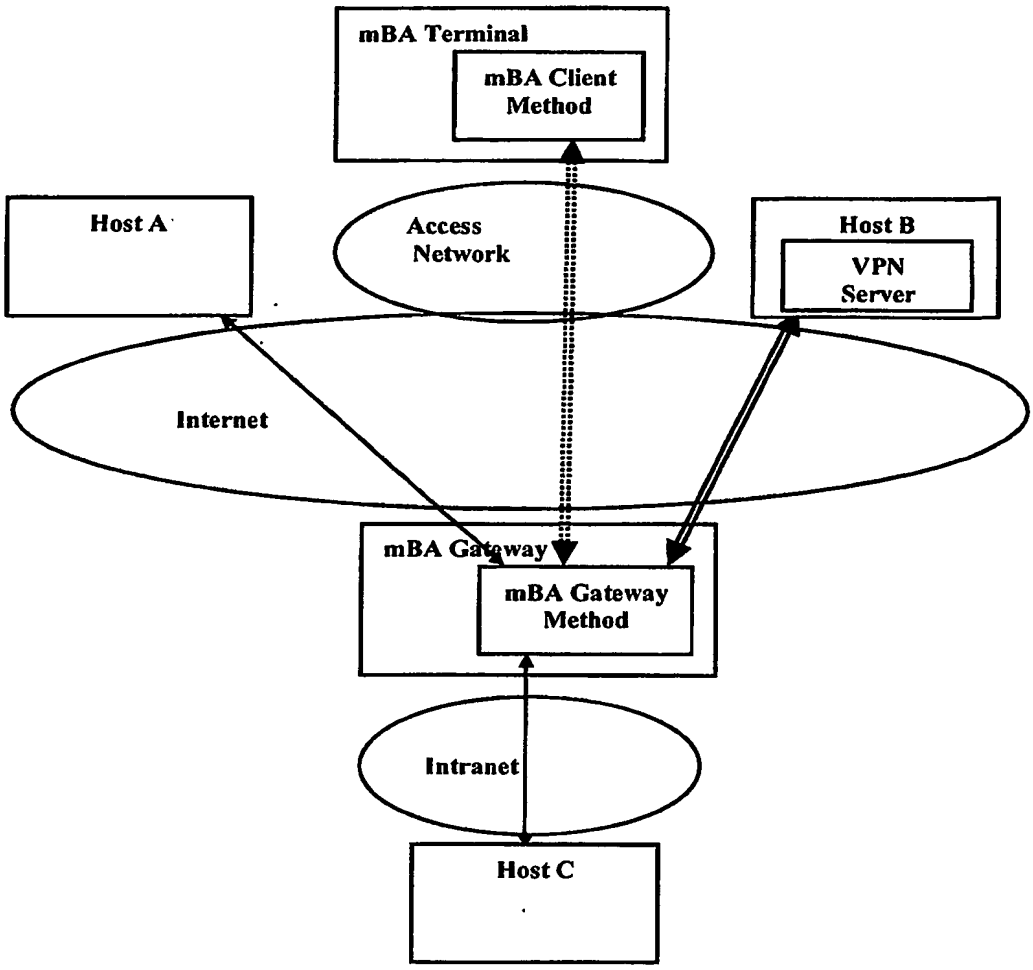

Terminal
VPN Client
Access Network
Internet
Host A
Firewall
VPN Gateway
Host B
VPN Server
Intranet
Host C
Legend
Secure VPN tunnel
Plain IP connection

Figure 1



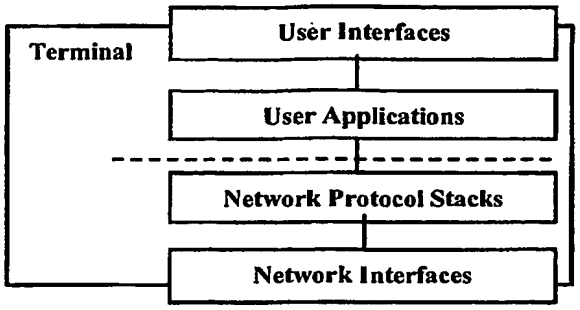
Legend

CCMPSS Tunnel

Plain IP connection

Secure VPN tunnel

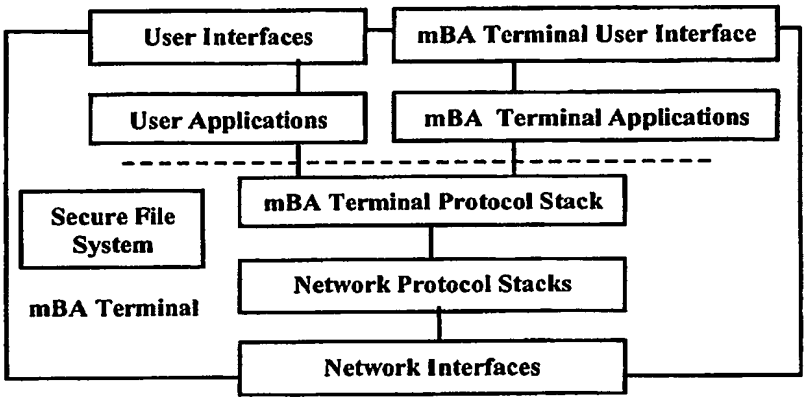
Figure 2



Legend

------ **Network API**

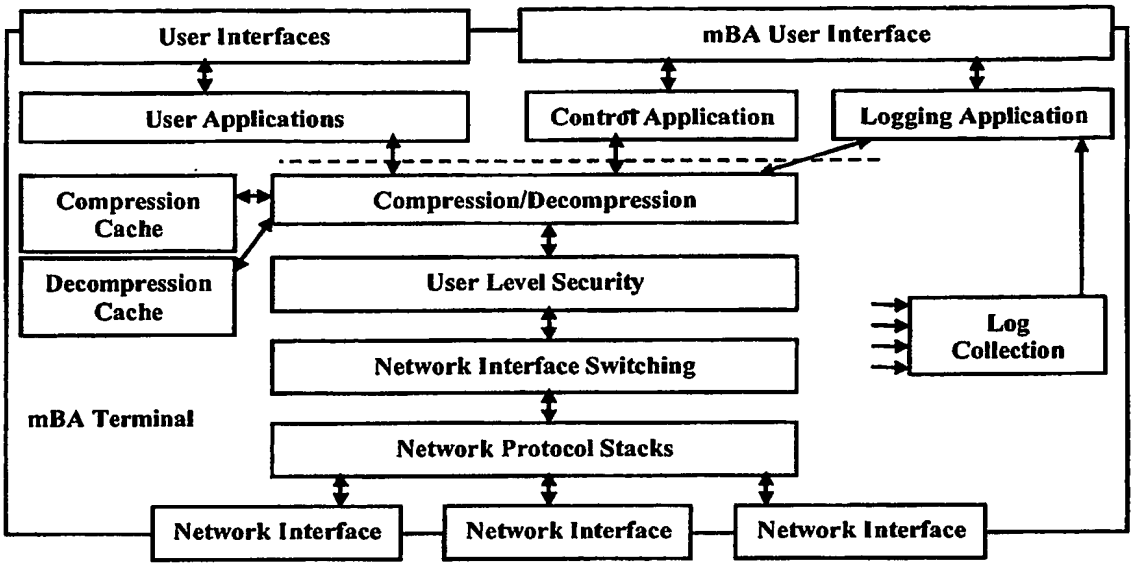
Figure 3



Legend

------ **Network API**

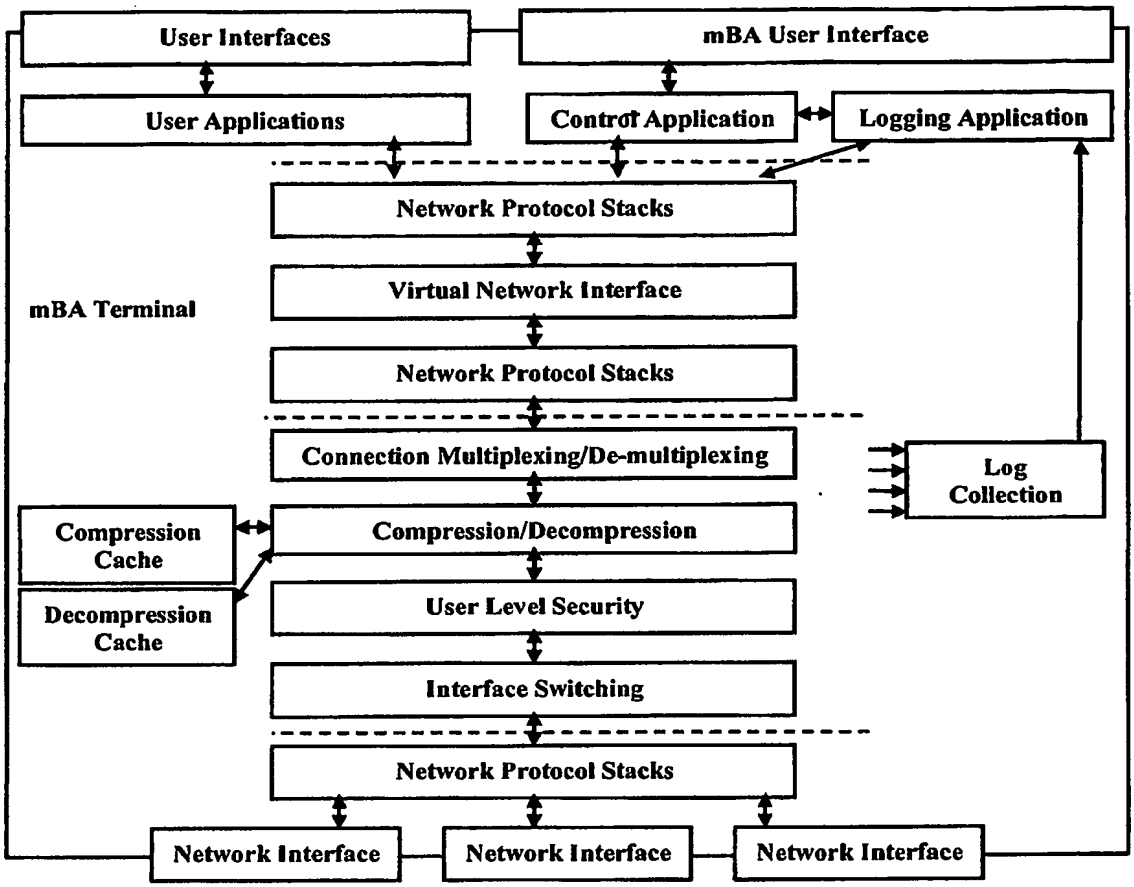
Figure 4



Legend

------ Network API

Figure 5



Legend

Network API

Figure 6

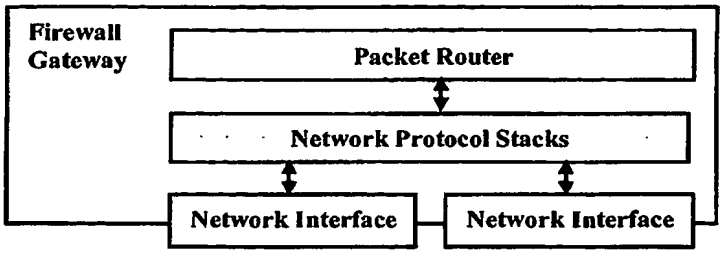
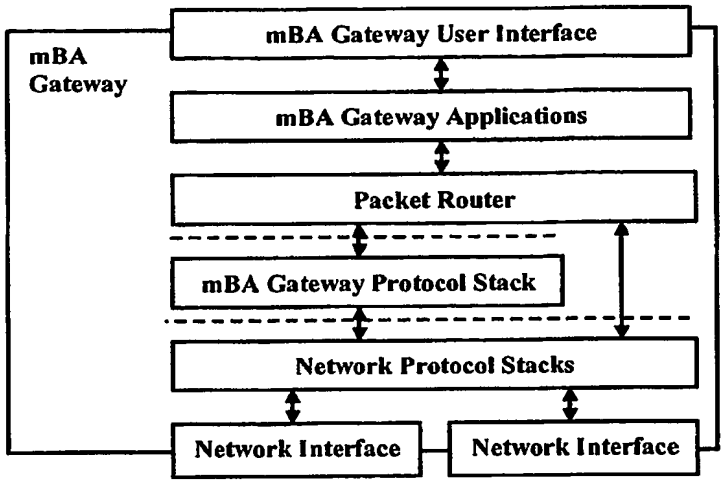

Firewall
Gateway
Packet Router
Network Protocol Stacks
Network Interface
Network Interface

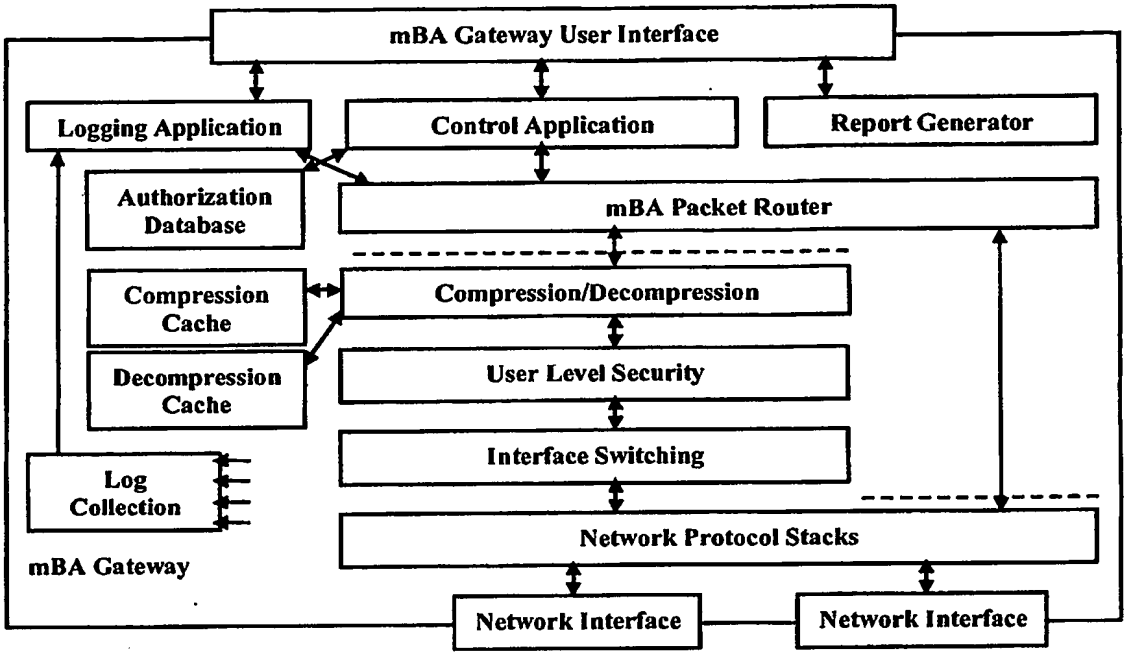
Figure 7



Legend

Network API

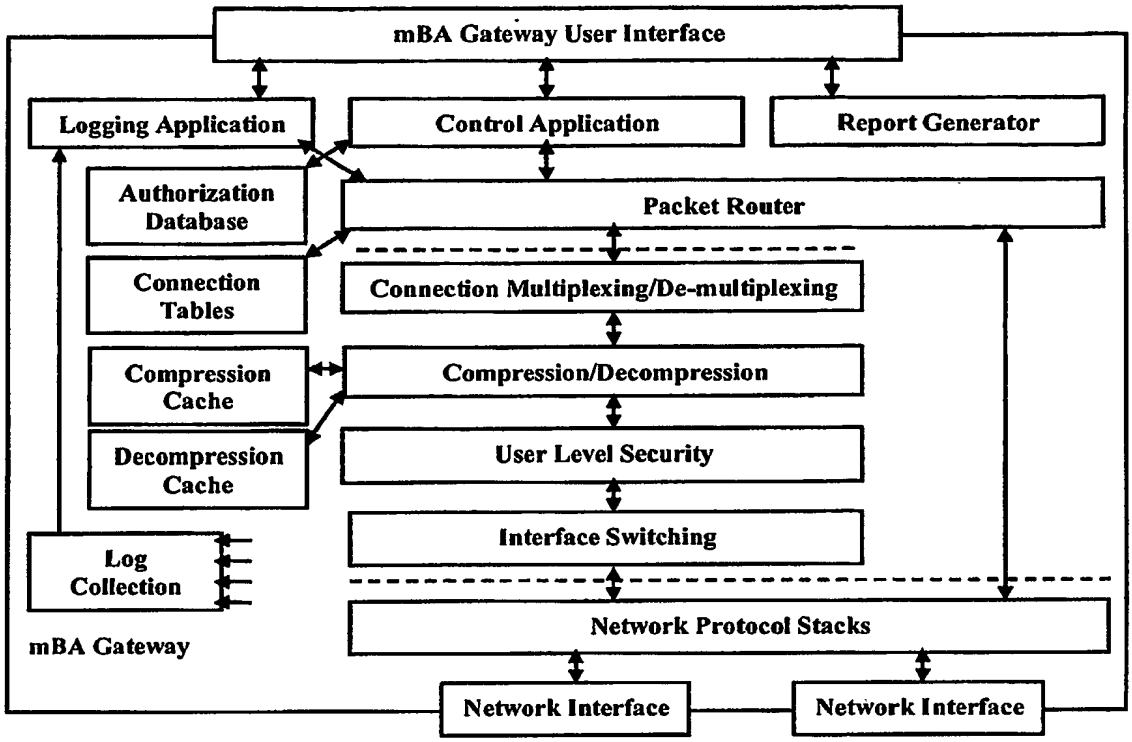
Figure 8



Legend

------ Network API

Figure 9



Legend

`------` **Network API**

Figure 10

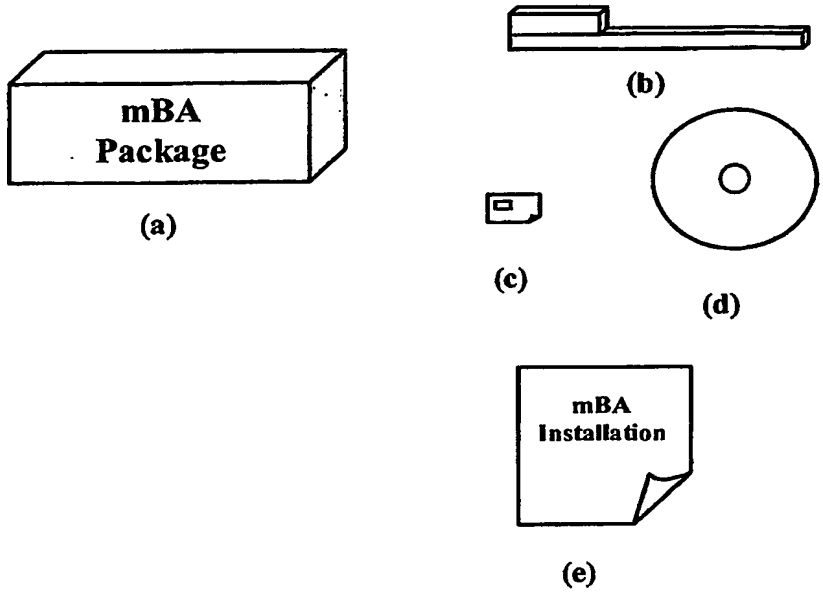
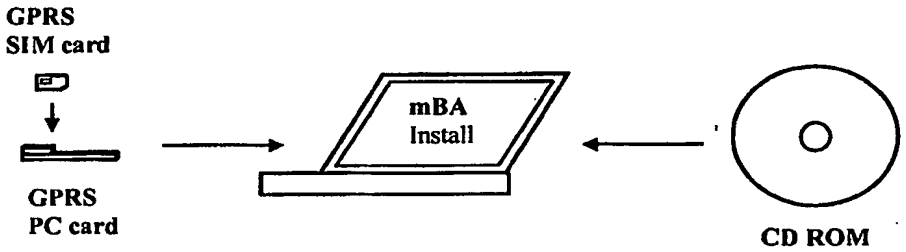
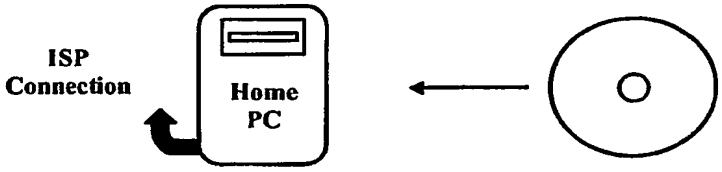


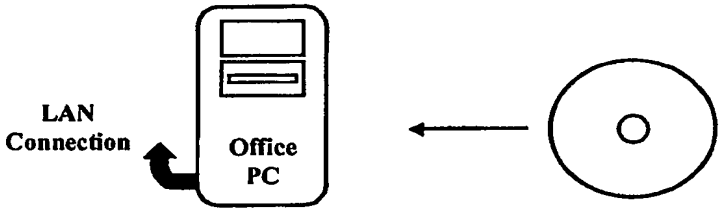
Figure 11



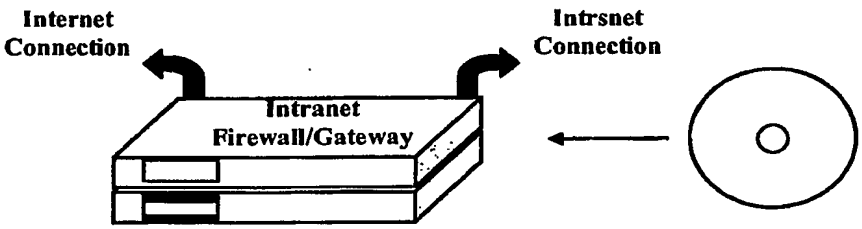
(a)



(b)



(c)



(d)

Figure 12